Informatics for Integrating Biology and the Bedside

i2b2 Functional Specification

Patient Count Plug-in

Document Version:	*1.2*
I2b2 Software Version:	*1.7*

Table of Contents

1. OVERVIEW

This plug-in calculates patient count breakdown for the children of a given concept. The patient count XML result is written to the **QT_XML_RESULT** table. To fetch this XML result, make a separate setfinder request with the result_instance_id.

1.1 Message

Run an Analysis plug-in by passing the plug-in name [CALCULATE_PATIENTCOUNT_FROM_CONCEPTPATH] and its parameters. The response message for this request is similar to the setfinder's run query request.

Request Type	**Request**	**Response**
CRC_QRY_runQueryInstance_fromAnalysisDefinition	analysis_definition	master_instance_result_responseType

1.1.1 Analysis Definition Type

Element Name	**Description**
analysis_plugin_name	Name of the plug-in. [CALCULATE_PATIENTCOUNT_FROM_CONCEPTPATH]
version	Plug-in version [1.0]
crc_analysis_input_param	This element contains the input xml that is defined in the parameter_info_xsd column of the qt_analysis_plugin table. ***Example*:** <crc_analysis_input_param name="ONT request"> <param type="int" column="item_key"> \\rpdr\RPDR\Diagnoses\Circulatory system (390-459)\ </param> </crc_analysis_input_param>
crc_analysis_result_list	This element contains the output xml that is defined in the parameter_info_xsd column of the qt_analysis_plugin table.

	<crc_analysis_result_list>
	    <result_output full_name="XML"
	        priority_index="1" name="XML"/>
	        . . .
	</crc_analysis_result_list>

1.1.2 Example Message

```
    <request_header>
        <result_waittime_ms>90000</result_waittime_ms>
    </request_header>

<message_body>
    <crc:psmheader>
        <request_type>
            CRC_QRY_runQueryInstance_fromAnalysisDefinition
        </request_type>
    </crc:psmheader>

<!- - Request message - ->
<crc:request  xsi:type="crc:analysis_definition_requestType">
    <analysis_definition>
        <analysis_plugin_name>CALCULATE_PATIENTCOUNT_FROM_CONCEPTPATH
        </analysis_plugin_name>
        <version>1.0</version>
        <crc_analysis_input_param name="ONT request">
            <param type="int" column="item_key">
                \\rpdr\RPDR\Diagnoses\Circulatory system (390-459)\
            </param>
        </crc_analysis_input_param>
        <crc_analysis_result_list>
            <result_output full_name="XML" priority_index="1" name="XML"/>
        </crc_analysis_result_list>
    </analysis_definition>
</crc:request>

<!- - Response message - ->
<crc:response xsi:type="crc:master_instance_result_responseType">
    <query_master>
        <query_master_id>0</query_master_id>
            <name>CALCULATE_PATIENTCOUNT_FROM_CONCEPTPATH</name>
            <user_id/>
            <group_id/>
            <create_date>2000-12-30T00:00:00</create_date>
            <request_xml/>
        </query_master>
        <query_instance>
            <query_instance_id>0</query_instance_id>
            <query_master_id>0</query_master_id>
            <user_id/>
            <group_id/>
            <batch_mode/>
            <start_date>2000-12-30T00:00:00</start_date>
            <end_date>2000-12-30T00:00:00</end_date>
            <query_status_type>
                <status_type_id>6</status_type_id>
                <name>COMPLETED</name>
                <description/>
            </query_status_type>
        </query_instance>
        <query_result_instance>
            <result_instance_id>0</result_instance_id>
            <query_instance_id>0</query_instance_id>
            <query_result_type>
                <result_type_id>3</result_type_id>
                <name>XML</name>
                <display_type>CATNUM</display_type>
                <visual_attribute_type>LH</visual_attribute_type>
                <description>Generic query result</description>
            </query_result_type>
            <set_size>0</set_size>
            <obfuscate_method/>
            <start_date>2000-12-30T00:00:00</start_date>
            <end_date>2000-12-30T00:00:00</end_date>
            <query_status_type>
                <status_type_id>6</status_type_id>
                <name>COMPLETED</name>
                <description>COMPLETED</description>
            </query_status_type>
        </query_result_instance>
    </crc:response>
</message_body>
```

2. INSTALLATION

The following outlines the steps to install the patient count plug-in.

2.1 Download and Extract XML Source Code

Steps to download and extract the plug-in, common and xml source code to a target area.

1. Set up a target source_directory.
2. Extract the plug-in and common source code to the target source_directory. The source directory should have the following three projects.
    a. edu.harvard.i2b2.server-common
    b. edu.harvard.i2b2.xml
    c. edu.harvard.i2b2.crcplugin.patientcount

2.2 Configure Patient Count Plug-in

Steps to configure the patient count plug-in.

1. Go to patient count plug-in project: *cd edu.harvard.i2b2.crcplugin.patientcount*
2. Setup the plugin property files location.
    a. Open *etc/spring/patientcount_application_directory.properties*.
    b. Setup the location under which the properties files will be copied.

```
    edu.harvard.i2b2.crcplugin.pb.applicationdir=
/opt/jboss/standalone/configuration/crcapp
```

    c. Open *etc/spring/edu.harvard.i2b2.crcplugin.pb.properties*.
    d. Setup the PM Cell URL.

```
    edu.harvard.i2b2.crcplugin.pb.ws.pm.url=
http://localhost:9090/i2b2/services/PMService/getServices
```

3. Setup the datasource lookup properties.

***Note:** This is similar to the CRC datasource lookup setup. For more detail on lookup datasource, please refer to the CRC Installation document.*

4. Specify the jdbc properties to locate the ‘CRC_DB_LOOKUP’ table in *etc/spring/PatientCountApplicationContext.xml*.

```
<bean id="PBBootStrapDS" class="org.apache.commons.dbcp.BasicDataSource" destroy-method="close">
    <property name="driverClassName" value=" oracle.jdbc.driver.OracleDriver"/>
    <property name="url" value="jdbc:oracle:thin:@localhost:1521:xe"/>
    <property name="username" value="i2b2hive"/>
    <property name="password" value="demouser "/>
</bean>
```

5. Update entry in *etc/spring/edu.harvard.i2b2.crcplugin.pb.properties* to reflect the above change.

```
edu.harvard.i2b2.crcplugin.pb.ds.lookup.datasource=PBBootStrapDS
edu.harvard.i2b2.crcplugin.pb.ds.lookup.servertype=ORACLE
edu.harvard.i2b2.crcplugin.pb.ds.lookup.schemaname=i2b2hive
```

6. Setup the **Ontology Cell** url

```
    edu.harvard.i2b2.crcplugin.pb.delegate.ontology.url=
http://localhost:9090/i2b2/services/OntologyService
```

7. Setup the PM service account details

   a. Create the CRC service user account in the **Project Management cell** and assign the ‘*Manager*’ role to the user. CRC uses this PM user in the Ontology and PM cell calls.

Note: *The CRC uses this PM user in the Ontology and PM cell calls.*

```
# # # # # # # # # # # # # # # # # # # # # # # # # # #
# CRC service account properties
# # # # # # # # # # # # # # # # # # # # # # # # # # #
edu.harvard.i2b2.crc.pm.serviceaccount.user=AGG_SERVICE_ACCOUNT
edu.harvard.i2b2.crc.pm.serviceaccount.password=demouser
```

   b. Copy each project’s datasource registered in *etc/jboss/crc-ds.xml* to *etc/spring/PatientCountApplicationContext.xml.* The bean id value will be ‘java:’ plus the name in the crc-ds.xml.

```
<bean id="java:QueryToolDemoDS" class="org.apache.commons.dbcp.BasicDataSource"
destroy-method="close">
    <property name="driverClassName" value="oracle.jdbc.driver.OracleDriver"/>
    <property name="url" value="jdbc:oracle:thin:@localhost:1521:XE"/>
    <property name="username" value="i2b2demodata"/>
    <property name="password" value="demouser"/>
 </bean>
```

2.3 Build and Deploy edu.harvard.i2b2.crcplugin.patientcount

The following are the steps to build and deploy edu.harvard.i2b2.crcplugin.patientcount.

If this has been deployed in a previous installation (eg. ONT), then there is no need to repeat this step.

1. Run the following commands:
    a. cd source_directory/edu.harvard.i2b2.server-common
    b. ant dist
    c. cd source_directory/edu.harvard.i2b2.crcplugin.patientcount

2. Setup the Apache Launcher environment.
    a. The apache launcher will help run the java class as a simple command line. All the dependent jar files location can be easily configured using the launcher confix.xml

    b. Open build.properties and setup the plug-in deployment directory.

    ```
    crcplugin.home=/opt/jboss-as-7.1.1.Final/standalone
    Jboss.home=/opt/jboss-as-7.1.1.Final
    ```

    c. ant setup_launcher_folder
    d. ant deploy

3. Register the plug-in to the CRC cell: Insert the plug-in's metadata to QT_ANALYSIS_PLUGIN table. The command_line and working_folder values will change based on your crcplugin.home values in build.properties.

```
insert into QT_ANALYSIS_PLUGIN(plugin_id, plugin_name, description, version_cd,
command_line, working_folder, status_cd,group_id) values
('1','CALCULATE_PATIENTCOUNT_FROM_CONCEPTPATH' ,
'CALCULATE_PATIENTCOUNT_FROM_CONCEPTPATH' , '1.0' , ' /opt/jboss-as-
7.1.1.Final/standalone/analysis_commons_launcher/bin/run_conceptpatient_breakdow
n.sh' , '/opt/jboss-as-7.1.1.Final/standalone/analysis_commons_launcher/bin','A',’@’);
```